REMARKS

With this amendment, claims 1, 3-16, 18-25, 27-33, and 37-40 are pending in the present application. Claims 1, 14, 16, and 30-33 have been amended to more particularly recite the claimed the invention. The amendments to claims 1, 14, 16, and 30-33 are supported in the specification at, for example, page 1, section l. 4 through page 9, section l. 27. Claim 41 has been added. Support for claim 41 can be found at, for example, page 43, section l. 11 through page 44, section l. 2 and Figure 17.

Applicants submit herewith a Request for Continued Examination. Thus continued examination of the present application as amended is hereby requested.

The Applicant has carefully and thoughtfully considered the Office Action and the comments therein. For the reasons given below, it is submitted that this application is in condition for allowance.

Examiner Interview

Applicants thank Examiner Ke for the in-person interview of September 11, 2008 and the Interview Summary of September 15, 2008. During the interview claim 1, the Office Action of June 26, 2008, and the cited prior art were discussed.

Rejections Under 35 USC § 103(a) – Coad in view of Little

On pages 2-15, the Office Action rejects claims 1, 3-5, 7-16, 18-20, 22-25 and 27-40 as being unpatentable over U.S. Patent No. 6,851,105 to Coad et al. (Coad) in view of U.S. Patent No. 7,047,518 to Little et al. (Little). Applicants respectfully traverse the rejection. Coad and Little are discussed first, followed by the Action's rejections of claims 1, 3-5, 7-16, 18-20, 22-25 and 27-40.

A. Coad

Coad discloses "methods and systems for generating, applying, and defining patterns for existing code in order to improve the design and efficiency of the existing code." Coad, col. 1, l.

28-30 and col. 6, ll. 42-47. Patterns for software development ensure sound code architecture, help resolve common, recurring software development problems, and allow developers to organize, document, and produce more maintainable software. Coad, col. 6, l. 58-67.

Methods and systems disclosed by Coad utilize a software development tool that generates a pattern instance and applies the pattern instance to a portion of existing code to improve the design of the existing code. Coad, col. 6, ll. 42-47. The software development tool can also be used to create a new pattern from existing code. Coad, col. 6, ll. 42-47.

When improving the design of existing code, the software development tool performs three steps. First, the software development tool receives an element type to distinguish the type of pattern that is to be created. Coad, col. 7, l. 51-53 and Figure 2. Second, the software development tool displays a list of pattern options that are applicable to the selected element type and from which the developer can choose. Coad, col. 10, l. 1-3 and Figure 2. Third, once the developer has selected a pattern, the software generation tool applies the pattern to the code. Coad, col. 24, l. 64-67.

In addition, the creation of patterns in existing source code may be depicted in a diagram. See, e.g., Coad, Figs. 9-11, diagrams 9022 and 1022; and Coad, col. 25, ll. 36-46.

B. Little

Little discloses "a system for integrated computer software application development and modeling." Little, Abstract. The system provides tools to assist a software engineer in the refinement of a model of a software application by folding existing application and database knowledge into the model of the software application. Little, col. 1, l. 64-67 and col. 9, l. 41-53. Particularly, "Design Patterns and Model Verification tools assist the software developer by performing much of the routine work necessary to complete a model [of a software application]." Little, col. 9, l. 53-56. For example, tools may apply a design pattern to the user's model of the software application in order to improve the software model's integration and efficiency. Little, col. 10 37-40. Once the software model is complete, "source code generation creates the implementation from the [software] model and helps keep the model and source code synchronized." Little, col. 9, l. 56-58

C. Rejections of claims 1, 3-5, 7-16, 18-20, 22-25 and 27-40.

Initially, Applicants note that **the Action addresses at least one feature that does not appear in claim 1** – namely "each embedded code relating to a characteristic of code to be generated from the graphical interface" – and is, therefore, not addressed in this response.

Regarding claim 1, Coad and Little, taken either singly or in any reasonable combination, do not disclose or render obvious the claimed invention for at least the following two reasons.

First, Coad does not disclose a "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations," as recited in amended claim 1. Instead, Coad teaches a "software development tool that generates [a] pattern instance." Coad, col. 6, ll. 42-47. The software development tool "applies a pattern to a portion of existing code to improve the design of the existing code." Coad, col. 6, ll. 42-47 (emphasis added). According to Coad, **the creation of patterns in existing source code may be depicted in a diagram**. See, e.g., Coad, Figs. 9-11, diagrams 9022 and 1022; and Coad, col. 25, ll. 36-46. Hence, Coad discloses a diagram representing the creation of patterns in existing source code, **which is neither the same as nor renders obvious** a "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations," as recited in amended claim 1. Therefore, Coad does not disclose "the graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations," as recited in amended claim 1.

Furthermore, Little does not overcome the failings of Coad. Instead, Little discloses a computer software development system which allows a software architect to develop a model or plan of the desired software application and to use this plan as a blueprint for subsequent software development. Little, col. 1, 28-30 and 64-67. **According to Little, a model is an abstraction of the essentials of a complex problem, arrived at by filtering out the nonessential details, thus making the problem easier to understand**. Little, col. 11, 1. 23-33. For example, Little discloses two types of models: a static model, which is described by the Use Case diagrams in the Use Case View, and a dynamic model which describes the system behavior and illustrates events from actors to systems. Little, col. 11, l. 53-67. Hence Little discloses developing **a model of a software application** to aid in future software development, **which is**

neither the same as nor renders obvious a "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations," as recited in amended claim 1. Therefore, Little does not disclose "the graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations," as recited in amended claim 1.

Second, Coad does not disclose "the graphical model being capable of simulation based on the equations," as recited in claim 1. Instead, Coad teaches a "software development tool that generates [a] pattern instance." Coad, col. 6, ll. 42-47. The software development tool "applies a pattern to a portion of existing code to improve the design of the existing code." Coad, col. 6, ll. 42-47 (emphasis added). According to Coad, **the creation of patterns in existing source code may be depicted in a diagram**. See, e.g., Coad, Figs. 9-11, diagrams 9022 and 1022; and Coad, col. 25, ll. 36-46. Hence, Coad discloses a diagram which represents the creation of patterns in existing source code, **which is neither the same as nor renders obvious** "the graphical model being capable of simulation based on the equations," as recited in claim 1. Therefore, Coad does not disclose "the graphical model being capable of simulation based on the equations," as recited in amended claim 1.

Furthermore, Little does not overcome the failings of Coad. Instead, Little discloses a computer software development system which allows a software architect to develop a model or plan of the desired software application and to use this plan as a blueprint for subsequent software development. Little, col. 1, 28-30 and 64-67. **According to Little, a model is an abstraction of the essentials of a complex problem, arrived at by filtering out the nonessential details, thus making the problem easier to understand.** Little, col. 11, 1. 23-33. For example, Little discloses two types of models: a static model, which is described by the Use Case diagrams in the Use Case View, and dynamic model which **describes** the system behavior and illustrates events from actors to systems. Little, col. 11, l. 53-67. Hence Little discloses developing a model of a software application to aid in future software development where the model is capable of illustrating or describing, **but not simulating**, events from actors to systems. The model of Little therefore is **neither the same as nor renders obvious** "the graphical model being capable of simulation based on the equations," as recited in amended claim 1. Therefore,

Little does not disclose "the graphical model being capable of simulation based on the equations," as recited in amended claim 1.

Therefore, Coad and Little, taken either singly or in any reasonable combination, fail to disclose or render obvious claim 1.

For reasons set forth above, Applicants believe that claim 1 is allowable and respectfully request that the above rejection of claim 1 be withdrawn and that claim 1 be allowed.

Dependent claims 2-13 and 40 depend on claim 1 and are believed to be allowable for at least the same reasons as above. Therefore, Applicants respectfully request that the above rejection of claims 2-13 and 40 be withdrawn and that claims 2-13 and 40 be allowed.

Independent claims 14, 16, and 30-33 recite subject matter similar to that recited in claim 1, which Applicants believe is allowable over Coad in view of Little. Therefore, Applicants believe claims 14, 16, and 30-33 are allowable for at least the reasons set forth above. Applicants respectfully request that the above rejection of claims 14, 16, and 30-33 be withdrawn and that claims 14, 16, and 30-33 be allowed.

Claim 14 recites "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations," which is similar to "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations" as recited in claim 1. Claim 14 recites "the graphical model being capable of simulation based on the equations," which is similar to "the graphical model being capable of simulation based on the equations" as recited in claim 1. Claim 14 recites "the code generation goal being used to generate embedded code from the graphical model, the code generation goal relating to a characteristic of the code to be generated from the graphical model," which is similar to "the at least one code generation goal being used to generate embedded code from the graphical model in a graphical modeling environment, the at least one code generation goal relating to a characteristic of the code to be generated from the graphical model" as recited in claim 1. Claim 14 recites "generating embedded code in accordance with the code generation goal," which is similar to "generating embedded code in accordance with the at least one code generation goal," as recited in claim 1.

Claim 16 recites "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations," which is similar to "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations" as recited in claim 1. Claim 16 recites "the graphical model being capable of simulation based on the equations," which is similar to "the graphical model being capable of simulation based on the equations" as recited in claim 1. Claim 16 recites "the acquired at least one code generation goal being used to generate embedded code from the graphical model, the acquired at least one code generation goal relating to a characteristic of the code to be generated from the graphical model," which is similar to "the at least one code generation goal being used to generate embedded code from the graphical model in a graphical modeling environment, the at least one code generation goal relating to a characteristic of the code to be generated from the graphical model" as recited in claim 1. Claim 16 recites "generating embedded code in accordance with the acquired at least one code generation goal," which is similar to "generating embedded code in accordance with the at least one code generation goal," as recited in claim 1.

Claim 30 recites "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations," which is similar to "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations" as recited in claim 1. Claim 30 recites "the graphical model being capable of simulation based on the equations," which is similar to "the graphical model being capable of simulation based on the equations" as recited in claim 1. Claim 30 recites "the acquired at least one code generation goal being used to generate embedded code from the graphical model, the acquired at least one code generation goal relating to a characteristic of the code to be generated from the graphical model," which is similar to "the at least one code generation goal being used to generate embedded code from the graphical model in a graphical modeling environment, the at least one code generation goal relating to a characteristic of the code to be generated from the graphical model" as recited in claim 1. Claim 30 recites "one or more instructions for generating embedded code in accordance with the acquired at least one code generation goal," which is similar to "generating embedded code in accordance with the at least one code generation goal," as recited in claim 1.

Claim 31 recites "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations," which is similar to "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations" as recited in claim 1. Claim 31 recites "the graphical model being capable of simulation based on the equations," which is similar to "the graphical model being capable of simulation based on the equations" as recited in claim 1. Claim 31 recites "the code generation goal being used to generate embedded code from the graphical model, the code generation goal relating to a characteristic of the code to be generated from the graphical model," which is similar to "the at least one code generation goal being used to generate embedded code from the graphical model in a graphical modeling environment, the at least one code generation goal relating to a characteristic of the code to be generated from the graphical model" as recited in claim 1. Claim 31 recites "one or more instructions for generating embedded code in accordance with the code generation goal," which is similar to "generating embedded code in accordance with the at least one code generation goal," as recited in claim 1.

Claim 32 recites "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations," which is similar to "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations" as recited in claim 1. Claim 32 recites "the graphical model being capable of simulation based on the equations," which is similar to "the graphical model being capable of simulation based on the equations" as recited in claim 1. Claim 32 recites "the at least one code generation goal being used to generate embedded code from the graphical model in a graphical modeling environment, the at least one code generation goal relating to a characteristic of the code to be generated from the graphical model," which is similar to "the at least one code generation goal being used to generate embedded code from the graphical model in a graphical modeling environment, the at least one code generation goal relating to a characteristic of the code to be generated from the graphical model" as recited in claim 1. Claim 30 recites "one or more instructions for generating embedded code in accordance with the at least one code generation goal," which is similar to "generating embedded code in accordance with the at least one code generation goal," as recited in claim 1.

Claim 33 recites "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations," which is similar to "graphical model representing a dynamic system having time-changing behavior modeled with differential, difference, and/or algebraic equations" as recited in claim 1. Claim 33 recites "the graphical model being capable of simulation based on the equations," which is similar to "the graphical model being capable of simulation based on the equations" as recited in claim 1. Claim 33 recites "a process for preparing a graphical model for a code generation process for creating code based on the graphical model and at least one code generation goal, wherein the at least one code generation goal relates to a characteristic of the code," which is similar to "the at least one code generation goal being used to generate embedded code from the graphical model in a graphical modeling environment, the at least one code generation goal relating to a characteristic of the code to be generated from the graphical model" as recited in claim 1. Claim 33 recites "wherein the computer program generates code in compliance with the at least one code generation goal," which is similar to "generating embedded code in accordance with the at least one code generation goal," as recited in claim 1.

Dependent claims 15, 18-25, 27-29 and 35-39 depend on claims 15, 18-25, 27-29 and 35-39, and therefore are believed to be allowable for at least the same reasons as above. Therefore, Applicants respectfully request that the above rejection of claims 15, 18-25, 27-29 and 35-39 be withdrawn and that claims 15, 18-25, 27-29 and 35-39 be allowed.

Added Claims

Claim 41 has been added. Claim 41 depends on claim 1 and is therefore allowable, at least, for being dependent from an allowable claim. Therefore, Applicants respectfully request that claim 41 be allowed.

CONCLUSION

All of the stated grounds of rejection have been properly traversed, accommodated, or rendered moot. Applicants therefore respectfully request that the Examiner reconsider all presently outstanding rejections and that they be withdrawn. Applicants believe that a full and complete reply has been made to the outstanding Office Action and, as such, the present application is in condition for allowance. If the Examiner believes, for any reason, that personal communication will expedite prosecution of this application, the Examiner is hereby invited to telephone the undersigned at the number provided.

Prompt and favorable consideration of this Amendment is respectfully requested.

Respectfully submitted,

Dated: 9/26/08

By _____

Michael A. Sartori, Ph.D.
Registration No.: 41,289
Kyle D. Petaja
Registration No.: 60,309
VENABLE LLP
P.O. Box 34385
Washington, DC 20043-9998
(202) 344-4000
(202) 344-8300 (Fax)
Attorney/Agent For Applicant

#980400v2